DOCUMENT RESUME

ED 448 710 IR 020 473

AUTHOR	Nunez Esquer, Gustavo; Sheremetov, Leonid
TITLE	Virtual Learning Spaces in the Web: An Agent-Based Architecture of Personalized Collaborative Learning Environment.
PUB DATE	1999-10-00
NOTE	7p.; In: WebNet 99 World Conference on the WWW and Internet Proceedings (Honolulu, Hawaii, October 24-30, 1999); see IR 020 454. Figures contain font that is too small to be legible.
PUB TYPE	Reports - Descriptive (141) -- Speeches/Meeting Papers (150)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	*Computer System Design; Computer Uses in Education; *Cooperative Learning; *Distance Education; Educational Technology; Higher Education; *Instructional Design; Models; *World Wide Web
IDENTIFIERS	Collaborative Learning; *Learning Environments; Web Based Instruction

ABSTRACT

This paper reports on the results and future research work within the paradigm of Configurable Collaborative Distance Learning, called Espacios Virtuales de Apredizaje (EVA). The paper focuses on: (1) description of the main concepts, including virtual learning spaces for knowledge, collaboration, consulting, and experimentation, a "multi-book" (i.e., a personalized electronic book), and use of agent-based tools; (2) agents in EVA, including the multi-agent environment of personal education (e.g., Internet and filtering, collaboration, personal advisor, evaluator, and group monitor agents), multi-agent planning, and agents for virtual laboratories; (3) a unifying framework for agents in EVA, including the Agent Communication Channel Router, Agent Management System, and Directory Facilitator; and (5) the LAN/ATM (Local Area Network/Asynchronous Transfer Mode) hardware platform developed for the EVA environment. (Contains 23 references.) (MES)

Virtual Learning Spaces in the Web: an Agent-Based Architecture of Personalized Collaborative Learning Environment

Gustavo Núñez Esquer
Computer Science Research Center, National Technical University, Mexico
gustavo@cic.ipn.mx

Leonid Sheremetov
Computer Science Research Center, National Technical University, Mexico
sher@cic.ipn.mx

Abstract: This article reports on the results and future research work within the paradigm of Configurable Collaborative Distance Learning, which we have named EVA. The article focuses on the description of the main concepts, multiagent architecture, implementation of particular agents, software and LAN/ATM hardware platform developed for the EVA environment. Prototypes of agents have been developed using VC++, JAVA, LALO, and JATLite for Unix and Windows platforms.

1. Introduction

Since their conception, knowledge-based learning environments have offered significant potential for fundamentally changing the educational process [Mark & Greer 95, Chan 96, Youngblut 94]. Nevertheless, despite of many expectations, few learning environments have made the difficult transition from the laboratory to the classroom, so the development of pedagogically sound tools has been the time challenge. Recent trends in Web-based learning environments involve collaborative work as embodied in Computer Supported Cooperative Work for human-computer interaction and Distributed Artificial Intelligence for learning. These resulted from the proliferation of networks ranging from local sites to the Internet. Computer Supported Cooperative Learning (CSCL) systems have been built on the lessons learned from the past approaches such as Computer-Based Training, Intelligent Tutoring Systems and Interactive Learning Environments. Perhaps their most important significance is the ability to offer a collaborative environment that utilizes the experiences of other students, different teaching strategies, the recorded group expertise and the capabilities of software tools to arrive at a particular convergent solution in a group fashion.

The investigation project *EVA (Espacios Virtuales de Aprendizaje* in Spanish - *Virtual Learning Spaces)* applies the CSCL methodology to obtain a new paradigm of the Configurable Personalized Collaborative Learning [Núñez et al. 98]. This project is dedicated to the research and development of pedagogic models and information technologies that provide spaces of knowledge, collaboration, consulting, and experimentation for supporting the learning activities of teams separated geographically.

Agent technology is the promising way to approach these problems. The notion of agents is the central part of contemporary learning environments, where they act as virtual tutors, virtual students or learning companions, virtual personal assistants that help students to learn, mine information, manage and schedule their learning activities [Müller et al. 97, Barros & Perkusich 96, Gordon & Hall 98]. The main purpose of our project is to develop models, architectures and multi-agent environment for collaborative learning and experimentation. The focus of this paper is the multiagent system (MAS) architecture of the EVA environment.

2. EVA concepts

The main concepts of EVA are the following:

1. The Virtual Spaces of knowledge, collaboration, consultation and experimentation as the collection of agents and conventional software components working over the knowledge domains.
2. A "Multi-book" concept, a personalized electronic book, generated by concatenating of selected units of learning material (ULM) and associated personal and group learning activities along the learning trajectory for each knowledge domain.
3. The intensive use of agent-based tools

- as virtual tutors, virtual students or learning companions, virtual personal assistants that help students,
- for mining information, managing, planning and scheduling learning activities, information retrieval and filtering, student assistance and tracking for evaluation of their intentions and performance, and
- to organize the workgroups and groups activities, and re-configure their work and knowledge spaces.

The conceptual architecture of EVA is structured into four essential knowledge elements, which are information deposits and sets of programs called Virtual Learning Spaces. These spaces are: *knowledge* - all the necessary information to learn, *collaboration* - real and virtual companions that get together to learn, *consulting* - the teachers or tutors (also real and virtual), who give the right direction for learning and consult doubts, and *experimentation* - the practical work of the students in virtual environment to obtain practical knowledge and abilities. To represent knowledge we have proposed a model based on the hierarchy of domains and concept graph representation of knowledge and learning activities. This model is used for learning trajectory planning, virtual space configuration and student tracking.

3. Agents in EVA

The core of the EVA environment consist of a number of components, composed of a set of deliberative and auxiliary agents, considered below. Agents form the intermediate layer between the users and virtual spaces (fig. 1).

3.1 Multiagent environment of personal education

The environment of collaborative education and smart personal assistance for tutors and students, consists of several agent types:

Internet search and filtering agent (A_{SF}): automatically searches for additional teaching material in the Internet, not yet located in the Knowledge Space, according to student's learning trajectory and through natural language searching tools (such as Clasitex [Guzmán 98]). Synthesizes information compiled of several sources about the same topic and filter redundant and repetitive information.

Collaboration agent (A_C): compares student's academic development and profile to form collaboration groups, requests help or information from other assistants and make collaborative decisions on how to select, integrate, and order information to be shared between the tutor and students or inside the students group.

Personal advisor (A_{PA}): suggests a personalized study plan to student according to its academic formation, interests, abilities and advances, and modify study plan, if it is necessary. Selects, integrates, and orders the information to study; advises for the learning problem solving, tells where to find learning material and tests, help the student to choose sources, suggests topics of thesis or projects, indicates to the student virtual labs, where it can make practices. This agent communicates with the search agent to find information according to the topic, it communicates with collaboration agent to collaborate with the rest of the group members. Also reminds a student about events, conferences, videoconferences, courses, tasks, tests, etc.

Evaluator agent (A_E): verifies periodically the learning advances, tries to find causes of misunderstanding, communicates with the advisor agent to re-organize the information to study.

Our prototype of learning community incorporates also a *Group Monitor agent (A_{GM})* and a *Learning companion* agent *(A_{LC})*. The most of agents have been implemented in JATLite package with rule-based inferencing capabilities, programmed in Jess [JATLite 98, Friedman-Hill 98]. All agents collaborate through the message passing mechanism in agent communication language (currently, KQML) [Finin et al. 97].

3.2 Multiagent planning

A learner navigates the virtual learning spaces by routes (*study plans*) suggested in an automatic manner by EVA. So, the purpose of the planning system is to design a particular learning trajectory for each student in the knowledge space, formed by ULM, which is organized in knowledge domains. At the next stage, personalized books, called Multibooks, are armed by concatenating of selected ULM along the learning trajectory for each knowledge domain. In the same way, groups of students with similar interests are arranged. Finally, student learning activities must be scheduled to satisfy temporal constraints.

Initial study plan and learning activities scheduling is generated on the bases (i) of student's initial knowledge in each area of Computer Science at the graduate level according to the model, proposed by the ACM [ACM 91], and detected, for example, at the admission exam stage, and (ii) his interests in terms of sub-specialty or separated courses from the area, which defines student's final state in the knowledge space. Initial student's knowledge is considered as initial conditions for the common domain.

To perform planning and scheduling within the MAS paradigm, we have proposed to associate a planning agent *(A_P)* with each domain model. When viewed from the perspective of the system goal, the global study plan appears as an AND-OR tree progressing from the system goal (at the root), down through goals and plans, to local plan fragments distributed among the agents. Constraints are associated with each node of the tree. Since local goals and decisions are interconnected, developed multistage negotiation algorithm provides means by which an agent can acquire enough knowledge to reason about the impact of his local decisions and modify its behavior accordingly to construct a globally consistent decision [Sheremetov & Núñez 99].

Learning Activities Planning System with Multistage Negotiation is implemented using JATLite. It is composed of *n* planning agents and a *coordinator agent,* which inherit their methods from the RouterClient class. Agents are implemented as JAVA applets, so they also inherit methods from the Applet and Frame (to support graphic interface) classes of JDK 1.2 package.

3.3 Agents for virtual laboratories

Even though the virtual learning environments has been studied from a pedagogical perspective in the last years, few researchers has formulated clear methodologies how to create Virtual Laboratories for to cooperation in the WWW [Rodríguez et al. 98, Forbus & Kenneth 96], even less based on Virtual Reality Modeling Language (VRML) scenarios [Lemays et al. 96].

The EVA project contemplate in an integrated way, the following key elements:

- Tele-presence, including the monitoring and smart remote control of systems.
- Computer aided experimentation (real and virtual).
- Assistance in performing the experiments that enables configuration, performing and explanation of results.
- Distributed and collaborative work environment, based on VR interfaces.

We've developed a CASE tool, called EasyVRML to assign more complex behaviors to the VRML scenarios in a more simple form then conventional tools [Quintero et al. 98] (fig. 2). VRML allows to change the object and agents features of a scene (shape, color, position, size, orientation, etc) in function of determined events through the script languages [Mitra et al. 95]. We have developed several applications for the experimentation space of "Distributed Intelligent Systems" and "Virtual Reality Design" courses, which allow the students to develop their agents, execute them and visualize their behavior in the virtual worlds. These agents include the following types: *Behavior Agent* (A_B), which visualization must be generated in virtual environment, *Interface Agent* (A_I) to generate a user's interface (a Web page) and to allow the user to control the *Behavior Agents,* and *Experimentation companion* (A_{EC}), which pertains to the learning companion type, discussed above.

Actually, the implementation scheme is based on the integration of four programming languages: (1) LALO like agent programming language, (2) VRML for the generation of the virtual environment and the graphical representation of these agents allowing to generate 3D representations of dynamical scenes through a browser, (3) VC++, resulting code of the compilation of LALO source programs, (4) Java programmed interface between VRML and VC++.

4. A Unifying Framework for Agents in EVA

Since there exist a great number of education technologies, pedagogical models and styles, as well as software technologies to be reflected in the architecture of potential learning environment, we need to be concerned about whether any software architecture has a possibility of outliving its designers and of providing a suitable foundation for unanticipated additions of significant new features. Current consensus on these issues is probably that the most robust unit of reusability is a "framework" making use of "objects" or "components" and agents [Bradshow et al. 97, Grimes & Potel 95, Smirnov & Sheremetov 99]. In EVA, we have prototyped various components of a framework, with agents providing dynamic coupling and interoperability between components using standard interfaces and data formats.

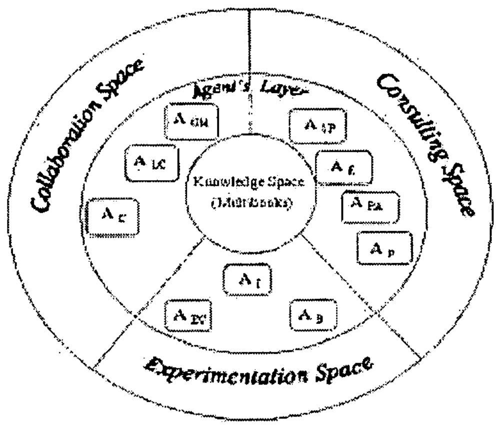

Figure 1. EVA conceptual architecture

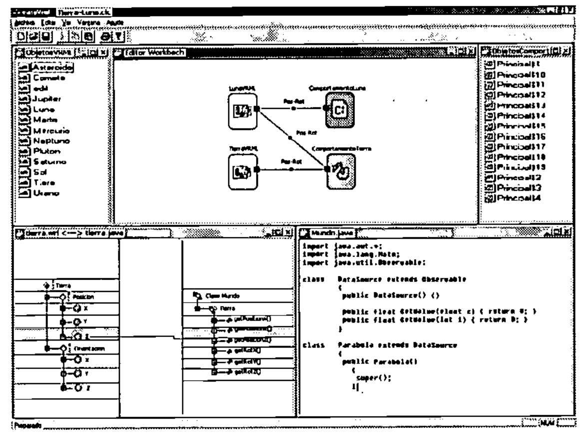

Figure 2. EasyVRML user interface

EVA agents are sorted into functional levels: presentation services, application services, management services, and data services. Agents providing presentation services are designed to hide the differences between viewers of different data types. From the point of view of the other agents, this means that there is a core viewer service protocol that is shared among all viewers. A viewer of a new kind of data need to implement an agent that converts this viewer service protocol to the specific call formats that the viewer application expects.

The application services level currently contains any agents supporting application services, discussed in the previous section of the article. Agent management services level is implemented according to the FIPA specification on the basis of the Microsoft DCOM model and includes the following components [FIPA 98, DCOM 96]:

- The Agent Communication Channel Router, which routes messages either to agents of the same platform or agents of different platforms.
- The Agent Management System, which coordinates the creation, erasing, suspension, resuming and authentication of agents in the platform.
- The Directory Facilitator, which holds a directory of services that offer each agent.

Finally, data services include *data locators* which encapsulate search and indexing functions, *data accessors* which retrieve data from heterogeneous data sources, and *data monitors* which feed information to clients based on user-configurable "push" policies.

Figure 3 shows the conceptual view of the overall EVA architecture, integrating principles of distributed object and agent technologies with the client-server architecture of the WEB. Specific client applications are built from various components that are integrated via an open presentation layer bus, such as Netscape's BeanConnect. The purpose of the bus is to allow HTML and client-side components (EVA agents, Java, JavaScript, plug-ins and ActiveX components, ORBs) to share a common object and messaging model, enabling seamless integration of tools, services, and user-interface elements.

Agent management services built on the foundation of existing distributed object services also allow agents the option of using a common agent-to-agent interlingua (currently, KQML) to communicate and coordinate their actions at the knowledge-level. In this case, EVA agents use an agent-to-agent (A2A) protocol that runs on top of standard lower-level protocols such as sockets or ORPC (for example, EVA planning and experimentation MAS). In addition to standard client-server connection protocols such as HTTP, RMI, and JDBC, a connection to a server-side DCOM component bus is provided and generic agent template with multi-level communication architecture, making use of KQML and ActiveX controls has been developed [Sheremetov & Smirnov 99]. This enables developers to selectively expose their interfaces, providing a standard way for system components to provide and access required services and data from each other. This not only ensures interoperability among our end-user tools and reusable components, but also allows us to take advantage of third-party DCOM and CORBA services that can be used and customized as needed.

The EVA Internet front-end (versión 1.0) with the above mentioned functionality is installed on WEB UNIX and WindowsNT servers Apache and Java Web Server. User interface in the knowledge space with personal advisor agent is depicted in fig. 4.

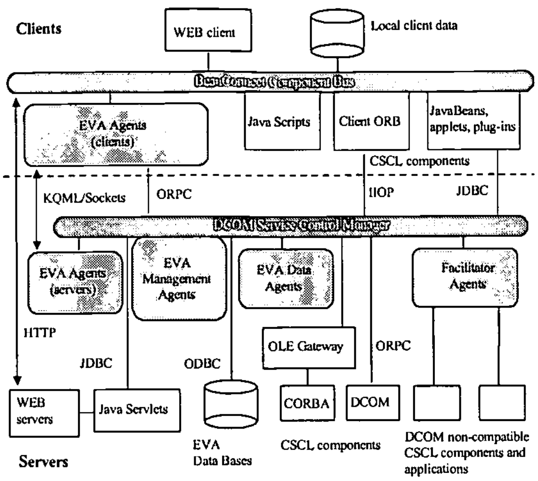

Figure 3. EVA Unifying Framework

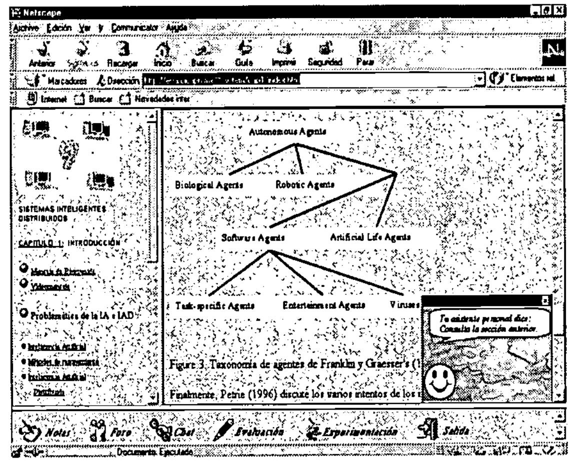

Figure 4. EVA user interface in the knowledge space

5. EVA platform

Developments like those proposed in this paper would not be possible without nowadays achievements of computer networking and telecommunications. So, we explore the advantages of the key communication technologies like Asynchronous Transfer Mode (ATM) and SONET (Synchronous Optical Network) as a transmission standard [Martinez & Su 99]. Then, to be able to test the EVA environment, the LAN/ATM based Experimental Distance Learning Laboratory was installed. The core of the system is an ATM switch with 8 gates at 155 Mbps, with the possibility to extend the capacity up to 622 Mbps, connected to the IPN ATM backbone and to an ATM LAN switch with 24 Ethernet inputs (10 Megabits each).

This Laboratory is the key element for the development of the ATM Experimental Center of Information Exchange, where the following on-going projects are being developed:

- Development, implementation and testing of teaching/learning activities in EVA environment
- Testing of performance of multimedia applications via assignment of virtual channels and bandwidth of the LAN/ATM network
- Transmission of signals and images over ATM using different compression techniques.

We also consider as the future work the integration of the Center into the Internet2 Mexican initiative, which is now at the planning stage and will be operative in 2001.

6. Conclusions

The EVA environment makes intensive use of agent technology, developing models, architectures and multi-agent environment for collaborative learning and experimentation, such as (i) multiagent generic open environment, based on federation architecture, models of knowledge sharing, and generic agent template with multi-level communication architecture, making use of KQML and ActiveX controls, (ii) personal learning assistants with information filtering capabilities (iii) agents for individual and collaborative learning with artificial learning companions, (iv) agents for planning of learning trajectories, and (v) multiagent experimentation space.

EVA is effectively used by the students of the CIC-IPN in the domain of the Master and Doctorate education in Computer Science. At the current stage of the experiment we are also developing the course on "The Potential of Virtual Reality in the Teaching of Engineering" to be delivered at the University level for about 200 teachers of engineering. We are working in a strict evaluation of the improvements brought by the mentioned techniques.

7. References

[ACM, 91] Computing as a Discipline, ACM report, 1991.

Barros Costa, E., & Perkusich, A.(1996). Modeling the Cooperative Interaction in a Teaching/Learning Situation. In *Intelligent Tutoring Systems*: 168-176.

Bradshaw, J. M., Dutfield, S., Benoit, P., & Woolley, J. D. (1997). KAoS: Toward an industrial-strength generic agent architecture. In J. M. Bradshaw (Ed.), *Software Agents*. Cambridge, MA: AAAI/MIT Press: 375-418.

Chan, T.W. (1996).Learning companion Systems, Social Learning Systems, and Intelligent Virtual Classroom, *In Proc. of the World Congress on AI in Education*, 1996, Vol. 7, No. 2, 125-159.

Distributed Component Object Model protocol DCOM/1.0 (1996), Network Working Group.

Finin, T., Labrou, Ya., & Mayfield J. (1997) KQML as an agent communication language, in J. Bradshaw, ed., Software Agents, AAAI/MIT Press, 1997.

FIPA7412, (1998). Draft 1.0, document of the Foundation for Intelligent Physical Agents (FIPA).

Forbus, D. & Kenneth, (1996). Articulate Virtual Laboratories for Science and Engineering Education; http://www.qrg.ils.nwu.edu/projects/NSF/avl.htm

Gordon, A. & Hall, L. (1998).Collaboration with Agents in a Virtual World. *In Proc. of the Workshop on Current Trends and Artificial Intelligence in Education, 4 World Congress on Expert Systems*, Mexico, 1998.

Grimes, J. & Potel, M. (1995). Software is headed toward object-oriented components. *IEEE Computer*(August), 24-25.

Guzmán A. (1998). Finding the main themes in a Spanish document. *Journal Expert Systems with Applications,* Vol. 14, No. 1/2, Jan/Feb. 139-148.

JATLite Beta Complete Documentation (1998), Stanford University, http://java.stanford.edu

Lemays, L., Murdock, K., & Couch J. (1996). 3D Graphics and VRML 2, *Sams Net.*

Mark, M.A., and Greer, J.E. (1995) The VCR tutor: Effective Instruction for Device Operation. *Journal of the Learning Sciences,* 4(2):209-246.

Martínez J. & Su, J. (1999) La investigación en redes por conmutación de celdas en el CIC IPN: resultados obtenidos e investigaciones en curso. *Int. Conference CIMAF'99*, March 22 - March 26, La Habana, Cuba.

Mitra, Yasuaki Honda, Kouichi Matsuda, Gavin Bell, Chee Yu, Chris Marrin (1995), Moving Worlds: behaviors for VRML, http://www.vrml.sgi.com/spec.html

Müller, J., Wooldridge, M. J. and Jennings, N. R., eds.(1997), Intelligent Agents III: Proc. of the 3rd Int. Workshop on Agent Theories, Architectures, and Languages, *Lecture Notes in Artificial Intelligence*, Vol. 1193, (Springer-Verlag, Berlin, 1997)

Nuñez, G., Sheremetov, L., Martínez, J., Guzmán, A., & Albornoz, A. (1998). The Eva Teleteaching Project - The Concept And The First Experience In The Development Of Virtual Learning Spaces. *In Proceedings of the 15th IFIP World Computer Congress "The Global Information Society on the Way to the Next Millennium",* Vienna and Budapest, 31 August - 4 September, Part II, 769-778.

Quintero, R., Armenta, A., Balladares, L. & Nuñez G. (1998). EasyVRML Herramienta CASE para la asignación de comportamiento a componentes de mundo VRML. *XI Congreso Nacional ANIEI*, Mexico, 1998 (in Spanish)

Rodrigues de Queiroz, L. et al. (1998). A Robotics and Computer Vision Virtual Laboratory, IEEE'98 In *Proceedings of AMC'98*, Coimbra, June29-July 1, 694-699.

Sheremetov, L. & Núñez, G. (1999). Multi-Stage Cooperation Algorithm and Tools for Agent-Based Planning and Scheduling in Virtual Learning Environment, The 1st International Workshop of Central and Eastern Europe on Multi-agent Systems (CEEMAS'99), 30th May-3rd June, St. Petersburg, Russia. (to appear)

Sheremetov, L.B. & Smirnov, A.V. (1999). Component Integration Framework For Manufacturing Systems Re-Engineering Based In Agents And Objects. *International Journal of Robotics and Autonomous Systems,* Elsevier, North-Holland, (to appear).

Youngblut, Ch. (1994) Government - Sponsored Research and Development Efforts in the Area of Intelligent Tutoring Systems. Institute for Defense Analyses, USA.

Acknowledgments

The EVA project is the effort of many people. We would particularly like to thank our colleagues from CIC-IPN for their determination and contribution to the system development, especially Jesús Martínez. We also thank the students of Doctorate and Master Programs for contributing ideas and talent to EVA programming, especially R. Peredo, L. Balladares, R. Quintero and A. Armenta.

NOTICE

REPRODUCTION BASIS

☒ This document is covered by a signed "Reproduction Release (Blanket) form (on file within the ERIC system), encompassing all or classes of documents from its source organization and, therefore, does not require a "Specific Document" Release form.

☐ This document is Federally-funded, or carries its own permission to reproduce, or is otherwise in the public domain and, therefore, may be reproduced by ERIC without a signed Reproduction Release form (either "Specific Document" or "Blanket").